Funnel Bandits

Vamsi K. Potluru*
J.P. Morgan AI Research

Branislav Kveton
Google Research

Sameena Shah
J.P. Morgan AI Research

Manuela M. Veloso
J.P. Morgan AI Research

ABSTRACT

Online learning with bandit feedback has been studied extensively in the past decades and has numerous applications in search, advertising, recommender systems, and hyperparameter optimization. The most common bandit setting is that of immediate feedback, where the reward of the arm is observed immediately after it is pulled. Unfortunately, this is unrealistic in many domains, such as marketing and advertising, where user conversions may take months. Such delays have a significant impact on the regret of classic bandit algorithms. In this work, we propose a funnel bandit, where the learning agent gets partial feedback as the user progresses through the marketing funnel. The arms are marketing policies and the agent is rewarded when the user converts, at the end of the funnel. An interesting structure of this problem is that a suboptimal arm can be identified from partial feedback. We propose practical UCB-like and posterior sampling algorithms for our problem, and analyze the former. Our analysis shows that the regret can be independent of the total delay in feedback. To the best of our knowledge, this is the first result of this kind. We also evaluate our methods empirically on synthetic datasets.

ACM Reference Format:
, Vamsi K. Potluru, Branislav Kveton, Sameena Shah, and Manuela M. Veloso. 2021. Funnel Bandits. In *Proceedings of KDD 21 (MARBLE Workshop, KDD 21).* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

A *stochastic multi-armed bandit* [4, 9, 10] is an online learning problem where an *agent* sequentially pulls arms with stochastic rewards. The agent maximizes its expected cumulative reward. It does not know the mean rewards of the arms in advance and learns them by pulling the arms. This results in the so-called *exploration-exploitation trade-off*: *explore*, and learn more about an arm; or *exploit*, and pull the best empirical arm. In practice, the *arm* may be an advertisement and its *reward* is a click of the user.

Optimism in the face of uncertainty [1, 4, 5] and *Thompson sampling (TS)* [3, 16] are near-optimal exploration algorithms for many important problem classes. However, when the rewards are delayed, the performance of these algorithms may no longer be optimal [7]. Consider the setting where we would like to measure user activity each week and want to maximize the probability of being active in each week of the month. Each week would correspond to one stage and there would be four stages in a month. The user is active or inactive in a stage. When the user is inactive, the user *drops off* the funnel. We maximimize the probability that the user gets to the end of the funnel.

Previous approaches have considered the delayed feedback setting but do not take the full problem structure into account [7, 12]. For many practical problems of interest, we have a funnel-like structure which is relatively well understood in their respective domains. Customers typically fall off the funnel as they transition between the stages with rates dependent on the target applications. We propose funnel bandits to solve this problem which has the following characteristics:

- The regret consists of a problem-dependent delay term which is determined by an *intermediate* stage of the arm. This is in sharp contrast to typical bandit algorithms which have a regret term that depends on the maximum delay corresponding to the wait time across all the stages [7].
- Experiments show the effectiveness of our approach in the case of medium to large delays.

2 SETTING

We study the following online learning problem. In round $t \in [n]$, a single user arrives and we choose an action for that user. We have K actions and refer to each action as an *arm*. The *pulled arm* in round t is denoted by $I_t \in [K]$. After the arm is pulled, the user enters a funnel and moves between its L stages. If the user is active in stage $j < L$, the user moves to stage $j + 1$.

The movement to stage $j + 1$ is delayed. When the user is active in the last stage L, the learning agent gets reward 1. If the user is inactive in some stage, the user falls off the funnel and the learning agent gets reward 0. In each stage, we observe whether the user is active or inactive. Note that first the user enters a stage j and after a delay we receive the corresponding activity status.

Our setting: The user t is active in stage j given arm i is a Bernoulli random variable $Z_{i,j,t} \sim \mathrm{Ber}(\bar{Z}_{i,j})$, where $\bar{Z}_{i,j}$ is the probability of being active in stage j given arm i. We assume that $Z_{i,j,t}$ is independent over rounds t. In this notation, the user moves through all stages and succeeds in the last one when $\prod_{j=1}^{L} Z_{i,j,t} = 1$. If we had assumed independence across the stages as well, the probability of this event would have factored as: $\mathbb{P}\left(\prod_{j=1}^{L} Z_{i,j,t} = 1\right) = \prod_{j=1}^{L} \mathbb{P}\left(Z_{i,j,t} = 1\right) = \prod_{j=1}^{L} \bar{Z}_{i,j}$.

*Corresponding author: vamsi.k.potluru@jpmchase.com

MARBLE Workshop, KDD 21, August 2021, Singapore

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

The activity of user t up to stage j given arm i is denoted by $Y_{i,j,t} = \min_{j'=1}^{j} Z_{i,j',t}$. The success of arm i up to stage j is denoted by $\mu_{i,j}$ and defined as $\mu_{i,j} = \mathbb{P}\left(Y_{i,j,t} = 1\right)$. Note that $Y_{i,j,t} \sim \text{Ber}(\mu_{i,j})$. The delay in the movement of user t to complete the first j stages is denoted by $w_j (\geq 0)$. For the rest of the paper, we will work exclusively with $Y_{i,j,t}$.

The success probability of arm i is denoted by μ_i and is equal to $\mu_{i,L}$. Concretely, lets say user t arrives at time t and enters stage 1 of arm 2 and receives reward $Y_{2,1,t}$ after a delay of w_1. If the user is active then they enter stage 2 and receives a cumulative reward $Y_{2,2,t}$ after a subsequent delay of $w_2 - w_1$. The goal of the learning agent is to learn the arm that maximizes the probability that the average user reaches the last stage and is active. Since the delays in feedback are independent of the pulled arm, this is maximized by arm $i_* = \arg\max_{i \in [K]} \mu_{i,L}$. We refer to this learning problem as a *funnel bandit*.

This is equivalent to minimizing the expected n-round regret given by:

$$R(n) = n\mu_{i_*} - \sum_{t=1}^{n} \mu_{I_t} \tag{1}$$

Note that in our setting we have $\mu_{i,j} \geq \mu_i$ and satisfies a monotonic property as a function of stage. It can be observed that the success probability can only drop as we transition across the funnel. We require weaker assumptions on our model for designing our bandit algorithms. In particular, we only need that $\mu_{i,j} \geq \mu_i$ for each arm $i \in [K], j \in [L]$ and this includes the funnel bandit as a special case of our formulation. In the next two sections, we will propose two novel algorithms corresponding to the upper confidence bound (`UCB1`) and Bayes modeling (`Bayes`) class of algorithms.

3 FUNNEL ALGORITHMS

We will describe our proposed algorithms for the funnel bandit problem.

3.1 FunUCB

Our proposed algorithm `FunUCB` is described in Algorithm 1. It operates in three stages broadly as follows. At each round t user t arrives. The corresponding rewards $Y_{:,:,t}$ for user t get realized but are not observed by the algorithm. Secondly, we compute the number of observations of arm i up to stage j in the first t rounds as follows:

$$T_{i,j,t} = \sum_{\ell=1}^{t} \mathbb{1}\{I_\ell = i\}\, \mathbb{1}\left\{\ell + w_j \leq t\right\} \tag{2}$$

The number of times arm i was selected in first t rounds is given by the first term. Second term ensures that were actually observed by the algorithm due to the delay constraint. Taking the two terms together, we can compute the number of times arm i composed of first j stages was observed in the first t rounds. The corresponding empirical mean of arm i up to stage j in the first t rounds can be computed as:

$$\hat{\mu}_{i,j,t} = \frac{\sum_{\ell=1}^{t} \mathbb{1}\{I_\ell = i\}\, \mathbb{1}\left\{\ell + w_j \leq t\right\} Y_{i,j,\ell}}{T_{i,j,t}} \tag{3}$$

Algorithm 1 FunUCB for Funnel Bandits

for $t = 1, \ldots, n$ **do**
  // Compute UCBs of the arms
  for $i = 1, \ldots, K$ **do**
    for $j = 1, \ldots, L$ **do**
      $U_{i,j,t} = \hat{\mu}_{i,j,t-1} + \sqrt{\frac{1.5 \log t}{T_{i,j,t-1}}}$, where $\hat{\mu}_{i,j,t-1}$, $T_{i,j,t-1}$ are defined in (3), (2), respectively
    end for
    $U_{i,t} = \min_{j \in [L]} U_{i,j,t}$
  end for
  // Compute best arm
  $I_t = \arg\max_{i \in [K]} U_{i,t}$
  Choose arm I_t and corresponding reward $Y_{I_t,:,t}$ for user t gets realized
end for

and finally, we compute the `UCB1` upper confidence bound of arm i up to stage j in round t as

$$U_{i,j,t} = \hat{\mu}_{i,j,t-1} + \sqrt{\frac{1.5 \log t}{T_{i,j,t-1}}}\,. \tag{4}$$

Thirdly, we obtain the UCB of each of K arms as the minimum of the UCB of each of the corresponding L stages $U_{i,t} = \min_{j \in [L]} U_{i,j,t}$. In particular, since we assume that $\mu_{i,j} \geq \mu_i$ for all i and j, any $U_{i,j,t}$ is a valid high-probability upper bound on μ_i. It makes sense to choose the most conservative one and this is what we essentially do.

`FunUCB` has some attractive properties which can be summarized as follows. We take advantage of the feedback of each of the stages and update our UCBs of each of the arms. A priori, it is not clear if this will lead to better regret performance but experimental results strongly show the benefits of such an approach. Also, we show upper bounds on the regret performance as a function of the delay feedback depending on the problem gap-dependent intermediate stage j.

3.2 Algorithm FunBayes

We also propose a novel Bayesian approach, namely `FunBayes` and it is described in Algorithm 2. Recently, the connection between posterior sampling and UCB algorithms has been established [15]. Similar to `UCB1` where we maintain upper confidence bounds on the arms, we maintain beta distributions for each of the stages of the arm and query a round-dependent quantile. The algorithm can be described as follows. Firstly, user t arrives at round t. Similarly to `FunUCB` the corresponding rewards $Y_{:,:,t}$ get realized but are not observed by the algorithm. The setting for the reward feedback and the corresponding delays are similar to `FunUCB`. Secondly, we compute the parameters of the beta distributions for arm i in stage j at time t as follows:

$$\alpha_{i,j,t} = \sum_{\ell=1}^{t} \mathbb{1}\{I_\ell = i\}\, \mathbb{1}\left\{\ell + w_j \leq t\right\} Y_{i,j,\ell} + 1 \tag{5}$$

$$\beta_{i,j,t} = \sum_{\ell=1}^{t} \mathbb{1}\{I_\ell = i\}\, \mathbb{1}\left\{\ell + w_j \leq t\right\} (1 - Y_{i,j,\ell}) + 1 \tag{6}$$

where $\alpha_{i,j,t}$ and $\beta_{i,j,t}$ are the counts of arm i in stage j by round t. The terms in the computation are the same as the ones in `UCB1`.

Algorithm 2 FunBayes for Funnel Bandits

for $t = 1, \ldots, n$ **do**
  // Compute UCBs of the arms
  for $i = 1, \ldots, K$ **do**
    for $j = 1, \ldots, L$ **do**
      $V_{i,j,t} = Q(1 - \frac{1}{t(\log n)^c}, \text{Beta}(\alpha_{i,j,t}, \beta_{i,j,t}))$ where $\alpha_{i,j,t}$, $\beta_{i,j,t}$ are defined in Equations 5, 6.
    end for
    $V_{i,t} = \min_{j \in [L]} V_{i,j,t}$
  end for
  //Compute best arm
  $I_t = \arg\max_{i \in [K]} V_{i,t}$
  Choose arm I_t and corresponding reward $Y_{I_t,:,t}$ for user t gets realized
end for

And finally, we compute the UCB's based on the round-dependent quantile of the estimated empirical distribution as follows:

$$V_{i,j,t} = Q(1 - \frac{1}{t(\log n)^c}, \text{Beta}(\alpha_{i,j,t}, \beta_{i,j,t})) \quad \forall j \in [L] \tag{7}$$

where $Q(q, \rho)$ corresponds to the quantile function associated with distribution ρ that satisfies $P_\rho(X \leq Q(t, \rho)) = q$. We use a horizon dependent term $(\log n)^c$ for analyzing the algorithm but in the experiments we set $c = 0$ without any issues similar to [8].

4 ANALYSIS

We analyze and provide regret bounds for both the algorithms FunUCB and FunBayes.

4.1 UCB Analysis

The estimated mean reward of arm i in stage j is a random quantity. We analyze its concentration below.

Lemma 1. *Let*

$$E_{i,j,t} = \left\{ \left| \hat{\mu}_{i,j,t-1} - \mu_{i,j} \right| \leq \sqrt{\frac{1.5 \log t}{T_{i,j,t-1}}} \right\}$$

be the event that the estimated mean reward of arm i in stage j and round t is "close" to its actual mean. Let

$$E_t = \bigcup_{i \in [K]} \bigcup_{j \in [L]} E_{i,j,t}$$

be the event that all events $E_{i,j,t}$ in round t occur and $\bar{E}_t$ be its complement. Then

$$\sum_{t=1}^{n} \mathbb{P}\left(\bar{E}_t\right) \leq \frac{\pi^2}{3} KL\,.$$

Proof. Fix arm i, stage j, round t, and the number of observations $s = T_{i,j,t-1}$. Then by Hoeffding's inequality

$$\mathbb{P}\left(\left| \hat{\mu}_{i,j,t-1} - \mu_{i,j} \right| \geq \sqrt{\frac{1.5 \log t}{s}} \right) \leq 2t^{-3}\,.$$

Now we apply the union bound and get

$$\begin{aligned} \sum_{t=1}^{n} \mathbb{P}\left(\bar{E}_t\right) &\leq \sum_{i=1}^{K} \sum_{j=1}^{L} \sum_{t=1}^{n} \sum_{s=1}^{t-1} \mathbb{P}\left(\bar{E}_{i,j,t}, T_{i,j,t-1} = s\right) \\ &\leq 2KL \sum_{t=1}^{n} t^{-2} \leq \frac{\pi^2}{3} KL\,. \end{aligned}$$

This concludes the proof. □

Now we bound the regret of FunUCB. The analysis is under the assumption that $\mu_{i,j} \in [0, 1]$ for all arms i and stages j.

Theorem 2. *Let arm 1 be a unique optimal arm, that is $\mu_1 > \mu_i$ for all suboptimal arms $i > 1$. Then the n-round regret of* FunUCB *is bounded as*

$$R(n) \leq \frac{\pi^2}{3} KL + \sum_{i=2}^{K} \Delta_i \min_{j \in [L]: \Delta_{i,j} > 0} \frac{6 \log n}{\Delta_{i,j}^2} + w_j\,,$$

where $\Delta_i = \mu_1 - \mu_i$ is the suboptimality gap of arm i and $\Delta_{i,j} = \mu_{1,L} - \mu_{i,j}$.

Proof. First, we decompose the regret as

$$\begin{aligned} R(n) &= \mathbb{E}\left[\sum_{t=1}^{n} \Delta_{I_t} \right] \leq \mathbb{E}\left[\sum_{t=1}^{n} \Delta_{I_t} \mathbb{1}\{E_t\} \right] + \sum_{t=1}^{n} \mathbb{P}\left(\bar{E}_t\right) \\ &\leq \mathbb{E}\left[\sum_{t=1}^{n} \Delta_{I_t} \mathbb{1}\{E_t\} \right] + \frac{\pi^2}{3} KL \\ &= \sum_{i=2}^{K} \Delta_i \mathbb{E}\left[\sum_{t=1}^{n} \mathbb{1}\{I_t = i, E_t\} \right] + \frac{\pi^2}{3} KL\,, \end{aligned}$$

where the last inequality follows from Lemma 1.

Now we fix suboptimal arm $i > 1$ and any stage j such that $\Delta_{i,j} > 0$. Roughly speaking, this is the stage where arm i can be distinguished from arm 1. Let

$$F_{i,j,t} = \left\{ T_{i,j,t-1} > \frac{6 \log n}{\Delta_{i,j}^2} \right\}$$

be the event that arm i is observed sufficiently often in stage j. Based on this event, the number of pulls of arm i can be bounded from above as

$$\begin{aligned} &\mathbb{E}\left[\sum_{t=1}^{n} \mathbb{1}\{I_t = i, E_t\} \right] \\ &\quad \leq \mathbb{E}\left[\sum_{t=1}^{n} \mathbb{1}\left\{I_t = i, E_t, F_{i,j,t}\right\} \right] + \frac{6 \log n}{\Delta_{i,j}^2} + w_j\,. \end{aligned}$$

The extra factor of w_j is because the observation in stage j is delayed by w_j rounds after the arm pull.

On events E_t and $F_{i,j,t}$, arm i cannot be pulled. This is because the UCB of arm i is bounded from above by that of arm 1, since

$$\begin{aligned} U_{i,t} \leq U_{i,j,t} &= \hat{\mu}_{i,j,t-1} + \sqrt{\frac{1.5 \log t}{T_{i,j,t-1}}} \\ &\leq \mu_{i,j} + 2\sqrt{\frac{1.5 \log t}{T_{i,j,t-1}}} < \mu_{i,j} + 2\sqrt{\frac{1.5 \Delta_{i,j}^2 \log t}{6 \log n}} \\ &< \mu_{i,j} + \Delta_{i,j} = \mu_{1,L} \leq U_{1,t}\,. \end{aligned}$$

Now note that the above derivation holds for any stage j such that $\Delta_{i,j} > 0$. Therefore, we have

$$\mathbb{E}\left[\sum_{t=1}^{n} \mathbb{1}\{I_t = i, E_t\}\right] \leq \min_{j \in [L]:\Delta_{i,j}>0} \frac{6\log n}{\Delta_{i,j}^2} + w_j\,.$$

Finally, we chain all inequalities and get

$$\begin{aligned} R(n) &\leq \sum_{i=2}^{K} \Delta_i \mathbb{E}\left[\sum_{t=1}^{n} \mathbb{1}\{I_t = i, E_t\}\right] + \frac{\pi^2}{3} KL \\ &\leq \frac{\pi^2}{3} KL + \sum_{i=2}^{K} \Delta_i \min_{j \in [L]:\Delta_{i,j}>0} \frac{6\log n}{\Delta_{i,j}^2} + w_j\,. \end{aligned}$$

This concludes the proof. □

4.2 Discussion

- If we set $j = L$ for all i, we get the same bound as UCB1 that would wait for the feedback until the very end. Thus we are always better than this approach.
- Generally speaking, if we choose any $j \neq L$ such that $\Delta_{i,j} > 0$, we have a smaller gap. Therefore, the term with $\Delta_{i,j}^2$ is larger than if we waited until the end. However, the other term w_j is smaller. Our algorithm automatically adapts to the best stage j for any arm i, that minimizes the sum of these terms.

4.3 Bayes-UCB

We show a sketch of the regret bound using the Bayes-UCB approach [8]. This is motivated by the structure of Theorem 2.

THEOREM 3. *Let arm 1 be a unique optimal arm, that is $\mu_1 > \mu_i$ for all suboptimal arms $i > 1$. Then the n-round regret of* FunBayes *is bounded as*

$$R(n) \leq \sum_{i=2}^{K} \Delta_i \min_{\{j \in [L]:D_{i,j}>0\}} \frac{1+\epsilon}{D_{i,j}} \log n + o_{\epsilon,c}(\log n) + w_j\,.$$

where Δ_i is defined in Theorem 2 and $D_{i,j} = d(\mu_{i,j}, \mu_{1,L})$. Note that $d(p,q)$ is the KL divergence betweeen distributions p and q.

PROOF. We briefly sketch out the steps. Similar to the FunUCB proof, we first bound the expected number of pulls to distinguish arm i in stage j from that of arm 1 in stage L. Let the number of pulls of arm i in stage j in n rounds be denoted by the random variable $G_{i,j}$. Based on Theorem 1 in [8], it can be bounded from above as follows:

$$\mathbb{E}\left[G_{i,j}\right] \leq \frac{(1+\epsilon)(\log n)}{D_{i,j}} + R_n(\epsilon, c)$$

where $R_n(\epsilon, c) = o(\log n)$, for every $\epsilon > 0$ and $c \geq 5$. We then consider all the stages j where $D_{i,j} > 0$ and add an additional number of pulls w_j that could occur due to the corresponding wait time of stage j:

$$\begin{aligned} \mathbb{E}\left[\sum_{t=1}^{n} \mathbb{1}\{I_t = i\}\right] &\leq \min_{j \in [L]:D_{i,j}>0} G_{i,j} + w_j \\ &\leq \min_{j \in [L]:D_{i,j}>0} \frac{(1+\epsilon)(\log n)}{D_{i,j}} + o_{\epsilon,c}(\log n) + w_j \end{aligned}$$

The final step of the proof follows from adding up the regret corresponding to all the sub-optimal arms i not equal to 1. □

Similar discussion of FunUCB apply to the FunBayes algorithm as well.

5 RELATED WORK

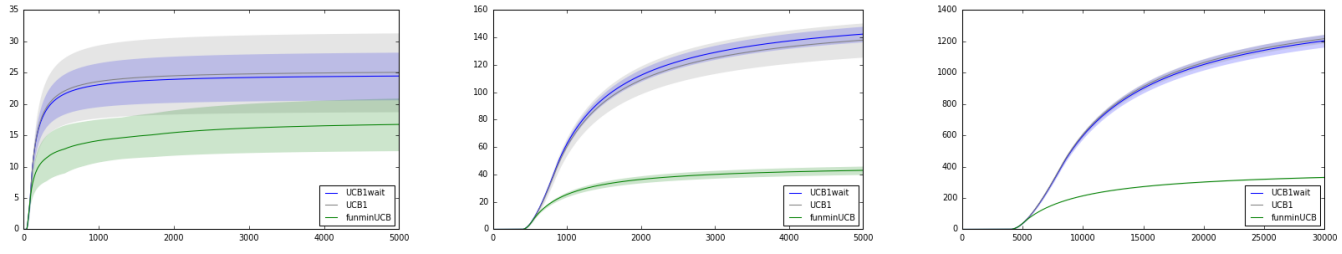
The problem of delayed feedback has received increased attention in the last couple of decades starting with [19]. They consider the adversarial full information setting with a fixed, known delay τ and show that the regret increases by a multiplicative factor of the delay. Later on, in [2] online stochastic optimization was considered and the regret was shown to increase as an additive factor of $\mathbb{E}\left[\tau^2\right]$ for i.i.d random delays. Similar results have been shown in bandit settings for fixed and constant delays in [6] by discounting a horizon dependent log factor. Around the same time, a multiplicative regret was established for the adversarial settings in [13]. [7] study how delayed feedback affects regret in online learning and provide additive regret bounds in the stochastic settings and multiplicative regret bounds in the adversarial settings. In comparison, we model the delayed feedback and leverage its structure to significantly reduce regret. In [12], the authors studied the problem of online prediction with delayed feedback. In particular, let x be the observed context, y be delayed feedback, and z be intermediate feedback. Then, under the assumption that $P(y \mid x) = \sum_z P(y \mid z)P(z \mid x)$ and that z is observed before y, learning of intermediate models $P(y \mid z)$ and $P(z \mid x)$ can lead to better predictions than directly learning $P(y \mid x)$. [11] study an online learning problem where the objective is to choose a waiting time to minimizes the total loss. This problem is only loosely related to our work. In particular, although our problem involves delays, we do not optimize them, and in fact cannot control them. [17] study the delayed feedback setting when the delays are stochastic and potentially censored. They also provide a lower bound and consider both UCB and UCB with KL confidence intervals. It would be interesting to extend our results in the censored setting. [14] study the delayed feedback setting when the feedback is aggregated as well. The regret is still shown to have an additive delay term as previous algorithms. Very recently, in [18] linear bandits have been tackled in the delay feedback model.

6 EXPERIMENTS

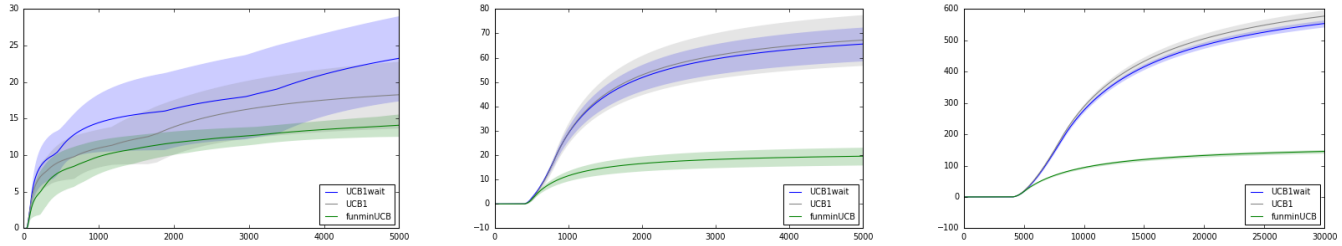
We evaluate the regret performance of our algorithms FunUCB and FunBayes.

In order to compare the efficacy of our proposed algorithms, we also consider the following delayed versions of UCB1 and TS for our settings. Let us first consider the approach for UCB1 and it can be similarly applied for TS. Instead of modeling the stages for each of the arms, we receive delayed reward for each of the selected arm based on the combined delay of all the stages. We call this the UCB1wait algorithm. We also consider the version where in the case of reward one, the algorithm waits for the sum of delays of all the corresponding stages, similarly to UCB1wait but for reward of zero it is immediately observed at the end of first stage of failure. Similarly, we have the corresponding versions of TSwait and TS algorithm. We will apply various bandit algorithms on synthetic datasets and study performance. Let us first consider the problem with two arms and four stages with various delay settings. The settings are as follows:

- (A) Arms 1 and 2 with Bernoulli success probabilities of 0.5 and 0.95 and delays in $\{10, 100, 100\}$ across 4 stages and 2

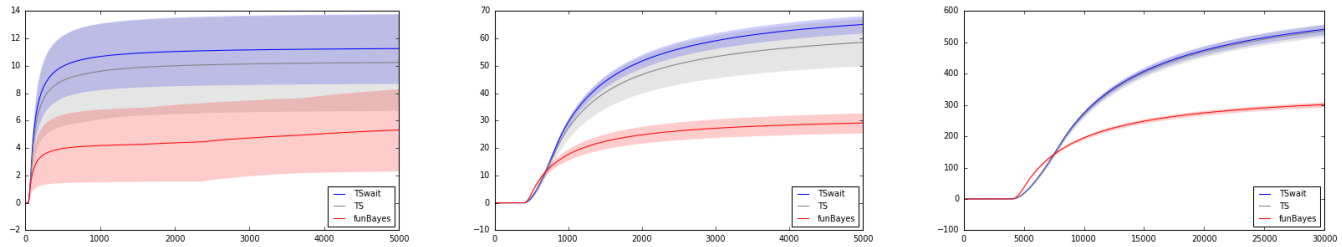


(a) Delay 10, $\Delta = 0.75$ **(b) Delay** 100, $\Delta = 0.75$ **(c) Delay** 1000, $\Delta = 0.75$



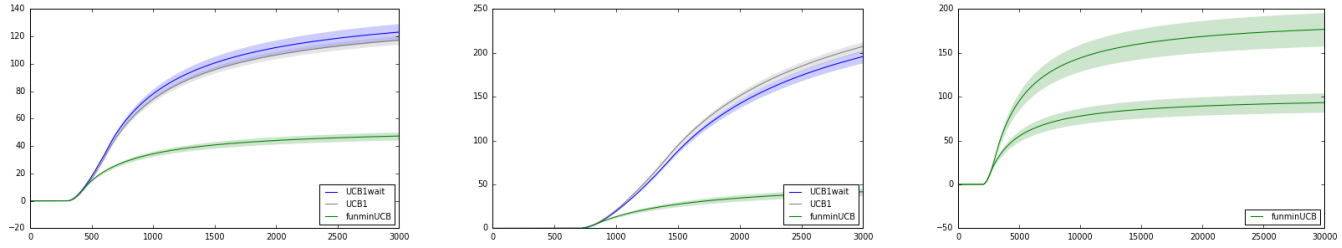
(d) Delay 10, $\Delta = 0.34$ **(e) Delay** 100, $\Delta = 0.34$ **(f) Delay** 1000, $\Delta = 0.34$

Figure 1: First note that FunUCB **(funminucb) has lower regret in all the delays considered in the experiments. We consider the various settings of two arms and four stages with delays of both** 10, 100 **and** 1000 **respectively. When the delays are large as in the case of** 100, 1000**, FunUCB outperform its corresponding naive approaches. Note that x-axis corresponds to rounds and y-axis to regret.**



(a) Delay 10, $\Delta = 0.34$ **(b) Delay** 100, $\Delta = 0.34$ **(c) Delay** 1000, $\Delta = 0.34$

Figure 2: FunBayes **has lower regret than** TS **and** TSwait**. The benefits are better for larger delays as can be seen by the plots.**



(a) Delay 100, 100, 100 **(b) Delay** 100, 100, 500 **(c)** 2, 4 **stages**

Figure 3: Plots (a), (b) show that our algorithm FunUCB **(aka funminUCB) has the same regret even though the delay has changed only for the last stage. However, for the other algorithms, the regret is much higher and depends on the full delay of all the stages. Last figure (c) shows the regret scales at most linearly with L for** FunUCB **(aka funminUCB). Note that x-axis corresponds to rounds and y-axis to regret.**

arms. This corresponds to $\mu_{1,j} = 0.5$, $\mu_{2,j} = 0.95$ for the two arms and $w_j \in \{10j, 100j, 1000j\}$.

- (B) Same as above but with success probabilities of the individual arms and stages replaced by Bernoulli success probabilities of 0.5 and 0.8 respectively.

Scaling with Δ*:* We observe that $\Delta = 0.75$ and $\Delta = 0.34$ for setting (A) and (B) respectively. Comparing the regret plots for delay of 1000 in Figure 1 corresponding to the two settings, we notice that the regret seems to scale linearly with Δ. and this is inline with the linear term corresponding to gap in Theorem 2. This can be observed by looking at the regret for delay 1000

Varying Delays: We vary the delays of the stages of the two arms and show the results for UCB version of the algorithm. All have two arms and four stages with delays in $\{10, 100, 1000\}$. The results are shown for `FunUCB` in Figure 1.

Confidence intervals: We compare the performance of `FunBayes` with `TS` and `TSwait` and across a wide range of delays, we see that we have a lower regret as seen in Figure 2.

Scaling with L:. We consider two arms with two stages and four stages. The delays are $[1000, 1000]$ for two stages and $[500, 500, 500, 500]$ for four stages. The gap between the arms is the same in both settings and is 0.34. We observe in Figure 3 that the regret scales at most linearly with L as bounded by our Theorem 2.

Total wait time $\gg w_j$*:* We consider three stages for the two arms with Bernoulli success probabilities as follows: $\mu_{1,1} = 0.5, \mu_{1,2} = 0.2, \mu_{1,3} = 0.1$ and $\mu_{2,j} = 0.95, \forall j \in [1, 2, 3]$. We consider two delay settings of $w_j = 100j$ and the other of $w_1 = 100, w_2 = 100, w_3 = 500$. The results are shown in Figure 3 and confirm that our regret does not scale with total wait time.

7 CONCLUSIONS AND OPEN PROBLEMS

We note that for all the runs, our algorithms `FunUCB` and `FunBayes` beats both `UCB1`, `UCB1wait` and `TS`, `TSwait` in terms of regret performance. We provide upper bounds on regret performance for both the UCB and Bayesian approach of our algorithms `FunUCB` and `FunBayes` respectively. Can we provide an extension of our algorithms to the funnel bandit problem with a contextual setting where each user has a context when they arrive? Note that our analysis is heavily dependent on being able to decouple the arms and relies mostly on counting arguments which is not possible in the contextual settings. For instance, playing sub-optimal arms can still provide information about other arms. Also, obtaining lower bounds for the funnel bandit problem would be of interest as has been established in other delayed settings.

REFERENCES

[1] Yasin Abbasi-Yadkori, David Pal, and Csaba Szepesvari. Improved algorithms for linear stochastic bandits. In *Advances in Neural Information Processing Systems 24*, pages 2312–2320, 2011.

[2] Alekh Agarwal and John C Duchi. Distributed delayed stochastic optimization. In *Advances in neural information processing systems*, pages 873–881, 2011.

[3] Shipra Agrawal and Navin Goyal. Analysis of Thompson sampling for the multi-armed bandit problem. In *Proceeding of the 25th Annual Conference on Learning Theory*, pages 39.1–39.26, 2012.

[4] Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, 47:235–256, 2002.

[5] Varsha Dani, Thomas Hayes, and Sham Kakade. Stochastic linear optimization under bandit feedback. In *Proceedings of the 21st Annual Conference on Learning Theory*, pages 355–366, 2008.

[6] Miroslav Dudik, Daniel Hsu, Satyen Kale, Nikos Karampatziakis, John Langford, Lev Reyzin, and Tong Zhang. Efficient optimal learning for contextual bandits. *arXiv preprint arXiv:1106.2369*, 2011.

[7] Pooria Joulani, Andras Gyorgy, and Csaba Szepesvári. Online learning under delayed feedback. In *International Conference on Machine Learning*, pages 1453–1461, 2013.

[8] Emilie Kaufmann, Olivier Cappé, and Aurélien Garivier. On bayesian upper confidence bounds for bandit problems. In *Artificial intelligence and statistics*, pages 592–600, 2012.

[9] T. L. Lai and Herbert Robbins. Asymptotically efficient adaptive allocation rules. *Advances in Applied Mathematics*, 6(1):4–22, 1985.

[10] Tor Lattimore and Csaba Szepesvari. *Bandit Algorithms*. Cambridge University Press, 2019.

[11] Tor Lattimore, András György, and Csaba Szepesvári. On learning the optimal waiting time. In *Algorithmic Learning Theory - 25th International Conference, ALT 2014, Bled, Slovenia, October 8-10, 2014. Proceedings*, volume 8776 of *Lecture Notes in Computer Science*, pages 200–214, 2014.

[12] Timothy Arthur Mann, Sven Gowal, Andras Gyorgy, Huiyi Hu, Ray Jiang, Balaji Lakshminarayanan, and Prav Srinivasan. Learning from delayed outcomes via proxies with applications to recommender systems. In *International Conference on Machine Learning*, pages 4324–4332, 2019.

[13] Gergely Neu, András György, Csaba Szepesvári, and András Antos. Online markov decision processes under bandit feedback. *IEEE Transactions on Automatic Control*, 59(3):676–691, 2013.

[14] Ciara Pike-Burke, Shipra Agrawal, Csaba Szepesvari, and Steffen Grunewalder. Bandits with delayed, aggregated anonymous feedback. In *International Conference on Machine Learning*, pages 4105–4113, 2018.

[15] Daniel Russo and Benjamin Van Roy. Learning to optimize via posterior sampling. *Mathematics of Operations Research*, 39(4):1221–1243, 2014.

[16] William R. Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3-4):285–294, 1933.

[17] Claire Vernade, Olivier Cappé, and Vianney Perchet. Stochastic bandit models for delayed conversions. In *Conference on Uncertainty in Artificial Intelligence*, 2017.

[18] Claire Vernade, Andras Gyorgy, and Timothy Mann. Non-stationary delayed bandits with intermediate observations. In *International Conference on Machine Learning*, pages 9722–9732. PMLR, 2020.

[19] Marcelo J Weinberger and Erik Ordentlich. On delayed prediction of individual sequences. *IEEE Transactions on Information Theory*, 48(7):1959–1976, 2002.